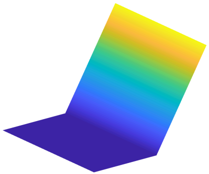
Regularizing Neural Networks via Radon-Domain Total Variation

Rahul Parhi
Biomedical Imaging Group
École polytechnique fédérale de Lausanne

MINDS Seminar

November 11, 2022



Neural Networks Outperform Everything

Deep neural networks are being used in many science and engineering problems, outperforming state-of-the-art methods.

- image classification,
- speech recognition,
- inverse problems in imaging,
- etc...

Big Caveat

They are poorly understood mathematically.

Fundamental Questions

- This raises the following fundamental questions:
  1. What kinds of functions do neural networks learn?
  2. Why can neural networks perform well in high dimensional settings?
  3. What is the right way to regularize a neural network?
- In this talk we will given (partial) answers to these questions.

Variational Formulation of Learning

Suppose that $f \in \mathcal{X}$, for some Banach space $\mathcal{X}$ on $\mathbb{R}^d$, and suppose we have the data $\{(\boldsymbol{x}_m, y_m)\}_{m=1}^{M} \subset \mathbb{R}^d \times \mathbb{R}$ generated from f.

- Consider the least-squares minimization problem

$$\min_{f \in \mathcal{X}} \sum_{m=1}^{M} |y_m - f(\boldsymbol{x}_m)|^2$$

$\implies$ This problem is **ill-posed**.

Question

How do we make this problem **well-posed**?

Answer

Regularize!

Regularization

Instead consider the minimization

$$\min_{f \in \mathcal{X}} \underbrace{\sum_{m=1}^{M} \ell(y_m, f(\boldsymbol{x}_m))}_{\text{data fidelity}} + \underbrace{\lambda |f|_{\mathcal{X}}^p}_{\text{regularization}},$$

where $|\cdot|_{\mathcal{X}}$ is a (semi)norm that defines $\mathcal{X}$.

- $\lambda > 0$ controls the strength of the regularization and the tradeoff between data fidelity and regularity.
- $\ell(\cdot, \cdot)$ is a **loss/error function**.

Classical Theory: Learning in Hilbert Spaces

Let $\mathcal{H}$ be a reproducing kernel Hilbert space (RKHS) and consider the variational problem

$$\min_{f \in \mathcal{H}} \sum_{m=1}^{M} \ell(y_m, f(\boldsymbol{x}_m)) + \lambda \|f\|_{\mathcal{H}}^2,$$

where $\ell(\cdot, \cdot)$ is convex. Then, the solution is unique and takes the form

$$f_{\mathsf{RKHS}} = \sum_{m=1}^{M} a_m k(\cdot, \boldsymbol{x}_m),$$

- $k(\cdot, \boldsymbol{x}_m)$ is the **reproducing kernel**: $\langle k(\cdot, \boldsymbol{x}_m), f \rangle_{\mathcal{H}} = f(\boldsymbol{x}_m)$.
- This is the well-known **representer theorem** for kernel methods. (de Boor and Lynch 1966; Kimeldorf and Wahba 1970)

Drawback of Hilbert Space Methods

- Hilbert spaces (e.g., L^2-Sobolev spaces) cannot efficiently capture functions that are spatially inhomogeneous or exhibit singularities.
- Hilbert space/kernel methods are **linear methods**, i.e.,

$\implies$ $T : (y_1, \ldots, y_M) \to f_{\mathsf{RKHS}}$ is a linear operator.

$\implies$ Linear methods are often suboptimal estimators.

$\implies$ Instead, consider **sparse** (nonlinear) methods.

Remark

This idea of considering sparse methods instead of L^2/Hilbert space methods is classical: wavelet shrinkage, LASSO, compressed sensing, etc.

Continuous-Domain Notion of Sparsity

- We have the finite-dimensional ℓ^1-norm:

$$\|\boldsymbol{u}\|_1 = \sup_{\substack{\boldsymbol{v}\in\mathbb{R}^d \\ \|\boldsymbol{v}\|_\infty = 1}} \boldsymbol{u}^\mathsf{T}\boldsymbol{v}.$$

- We have the infinite-dimensional ℓ^1-norm

$$\|u\|_{\ell^1(\mathbb{Z})} = \sup_{\substack{v\in c_0(\mathbb{Z}) \\ \|v\|_{\ell^\infty(\mathbb{Z})} = 1}} \sum_{n\in\mathbb{Z}} u[n]v[n]$$

- We have the continuous-domain analogue of the ℓ^1-norm

$$\|u\|_{\mathcal{M}(\mathbb{R}^d)} = \sup_{\substack{v\in C_0(\mathbb{R}^d) \\ \|v\|_{L^\infty(\mathbb{R}^d)} = 1}} \langle u, v\rangle$$

$\Longrightarrow$ $\mathcal{M}(\mathbb{R}^d) = \left(C_0(\mathbb{R}^d)\right)'$ is the space of finite Radon measures.
$\Longrightarrow$ $\mathcal{M}(\mathbb{R}^d)$ is the continuous-domain analogue of ℓ^1 (not $L^1(\mathbb{R}^d)$!).

What is $\mathcal{M}(\mathbb{R}^d)$?

- "Generalization" of $L^1(\mathbb{R}^d)$:

$\Longrightarrow$ $L^1(\mathbb{R}^d) \overset{\text{iso.}}{\hookrightarrow} \mathcal{M}(\mathbb{R}^d)$, i.e., for $f \in L^1(\mathbb{R}^d)$, $\|f\|_{L^1} = \|f\|_{\mathcal{M}}$.
$\Longrightarrow$ The inclusion $L^1(\mathbb{R}^d) \subset \mathcal{M}(\mathbb{R}^d)$ is **strict.**
$\Longrightarrow$ $\delta(\cdot - \boldsymbol{x}_0) \notin L^1(\mathbb{R}^d)$
$\Longrightarrow$ $\delta(\cdot - \boldsymbol{x}_0) \in \mathcal{M}(\mathbb{R}^d)$ with $\|\delta(\cdot - \boldsymbol{x}_0)\|_{\mathcal{M}} = 1$.

- Recovers the ℓ^1-norm since

$$\left\| \sum_{n=1}^{N} a_n \delta(\cdot - \boldsymbol{x}_n) \right\|_{\mathcal{M}} = \sum_{n=1}^{N} |a_n| = \|\boldsymbol{a}\|_1.$$

Comparing Hilbert Space vs. Sparse Methods

Consider the following function spaces defined in terms of the second (distributional) derivative of a function f, $\mathrm{D}^2 f$.

$$H^2[0,1] \coloneqq \left\{ f : [0,1] \to \mathbb{R} : \ \mathrm{D}^2 f \in L^2[0,1] \right\},$$
$$\mathrm{BV}^2[0,1] \coloneqq \left\{ f : [0,1] \to \mathbb{R} : \ \mathrm{D}^2 f \in \mathcal{M}[0,1] \right\}.$$

- The second-order L^2-Sobolev space H^2 is an RKHS.
- The second-order bounded variation space BV^2 is a Banach space with a **sparsity-promoting** norm.
  $\implies \ f \in \mathrm{BV}^2[0,1] \iff \mathrm{D} f \in \mathrm{BV}[0,1]$.
- $H^2[0,1] \overset{\text{c.}}{\hookrightarrow} \mathrm{BV}^2[0,1] \overset{\text{c.}}{\hookrightarrow} L^2[0,1]$, where the inclusions are strict.

A Learning/Recovery Problem

Suppose we want to learn/recover $f : [0,1] \to \mathbb{R}$ from the data

$$y_m = f(x_m) + \varepsilon_m,\ m = 1, \ldots, M,$$

where x_m are nicely distributed on $[0,1]$ (e.g., uniformly at random or equally spaced) and $\varepsilon_m \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$. We will discuss three techniques for learning this function:

- Cubic smoothing spline, a kernel/linear method;
- Linear locally adaptive spline, a sparse/nonlinear method;
- Wavelet shrinkage with Db3 wavelets, a sparse/nonlinear method.

Cubic Smoothing Splines

$$f_{M,\mathsf{sspl}} = \underset{f:[0,1]\to\mathbb{R}}{\arg\min} \sum_{m=1}^{M} |y_m - f(x_m)|^2 + \lambda \left\| \mathrm{D}^2 f \right\|_{L^2}^2$$

- Solution is **unique**.
- It is a **cubic spline** with knots at $\{x_m\}_{m=1}^{M}$.
- Representer theorem in an RKHS.

Carl de Boor and Robert E. Lynch (1966). "On splines and their minimum properties". In: **Journal of Mathematics and Mechanics** 15.6, pp. 953–969.

George S. Kimeldorf and Grace Wahba (1970). "A correspondence between Bayesian estimation on stochastic processes and smoothing by splines". In: **The Annals of Mathematical Statistics** 41.2, pp. 495–502.

Linear Locally Adaptive Splines

$$f_{M,\mathsf{las}} \in \operatorname*{arg\,min}_{f:[0,1]\to\mathbb{R}} \sum_{m=1}^{M} |y_m - f(x_m)|^2 + \lambda \underbrace{\|\mathrm{D}^2 f\|_{\mathcal{M}}}_{=:\, \mathrm{TV}^2(f)}$$

- Solution set is nonempty, convex, and weak* compact.
- **Extreme points** of solution set are **linear splines** with **adaptive** knot locations $\{t_n\}_{n=1}^{N}$ with $N < M$.
- Full solution set is convex hull of extreme points.

$\Longrightarrow$ Solution set is completely characterized by sparse linear splines.

- Representer theorem in a **Banach space**.

Stephen D. Fisher and Joseph W. Jerome (1975). "Spline solutions to L^1 extremal problems in one and several variables". In: **Journal of Approximation Theory** 13.1, pp. 73–83.

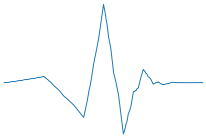
Enno Mammen and Sara van de Geer (1997). "Locally adaptive regression splines". In: **The Annals of Statistics** 25.1, pp. 387–413.

Michael Unser et al. (2017). "Splines Are Universal Solutions of Linear Inverse Problems with Generalized TV Regularization". In: **SIAM Review** 59.4, pp. 769–793.

Db3 Wavelet Shrinkage

$$\alpha_{M,\mathsf{wav}} \in \underset{\alpha[\cdot]\in\ell^1(\mathbb{Z})}{\arg\min} \sum_{m=1}^{M} |y_m - f_\alpha(x_m)|^2 + \lambda\|\alpha\|_{\ell^1(\mathbb{Z})},$$

- Impose that $f_\alpha = \sum_{n\in\mathbb{Z}} \alpha[n]\psi_n$.



- $\{\psi_n\}_{n\in\mathbb{Z}}$ is ordering of the Db3 wavelet basis on $[0, 1]$.
  $\Longrightarrow$ Translates and dilates of the mother wavelet.
- $f_{M,\mathsf{wav}} := f_{\alpha_{M,\mathsf{wav}}}$.

David L. Donoho and Iain M. Johnstone (1998). "Minimax estimation via wavelet shrinkage". In: **The Annals of Statistics** 26.3, pp. 879–921.

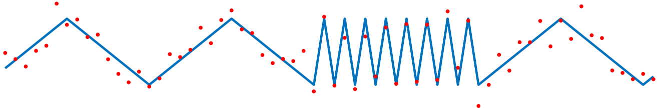
Performance

- Measure the performance of an estimator f_M for f by the **mean-squared error**: $\mathbb{E}\|f - f_M\|_{L^2}^2$.

	$f_{M,\text{sspl}}$	$f_{M,\text{wav}}$	$f_{M,\text{las}}$
$f \in H^2[0,1]$	$M^{-4/5}$	$M^{-4/5}$	$M^{-4/5}$
$f \in \mathrm{BV}^2[0,1]$	$M^{-3/4}$	$M^{-4/5}$	$M^{-4/5}$

- Remarks...
  - $\Longrightarrow$ The **minimax rates** for $H^2[0,1]$ and $\mathrm{BV}^2[0,1]$ are $M^{-4/5}$.
  - $\Longrightarrow$ The smoothing spline estimator (or any linear estimator) is suboptimal for BV^2 functions. (Donoho and Johnstone 1998)
  - $\Longrightarrow$ No estimators can perform better than the (nonlinear) wavelet shrinkage or locally adaptive spline estimators for BV^2 functions.

An Example



Generate noisy data $\{(x_m, y_m)\}_{m=1}^{M}$ from $f \in \mathrm{BV}^2[0,1]$:

$$y_m = f(x_m) + \varepsilon_m,\ m = 1, \ldots, M$$

- $f \notin H^2[0,1]$.
- f is **spatially inhomogeneous**.

Linear Methods: Cubic Smoothing Splines

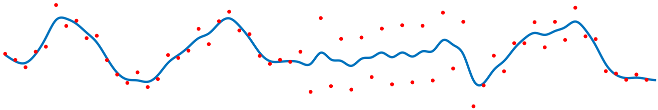


Figure: cubic smoothing spline, large λ

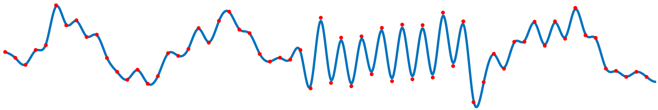


Figure: cubic smoothing spline, small λ

- Smoothing spline either **oversmooths** high variation portion of data or **undersmooths** low variation portion of data.
- Smoothing splines **cannot adapt** to the inhomogeneity of the underlying function.

Nonlinear Methods: Wavelets and Adaptive Splines

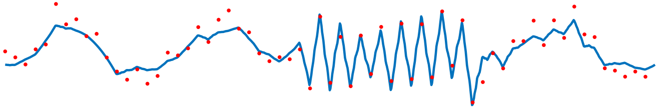


Figure: Db3 wavelet shrinkage

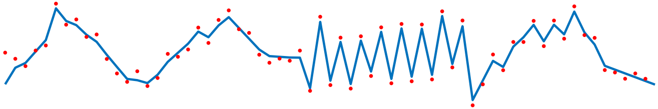


Figure: Linear locally adaptive spline

- Wavelet shrinkage and locally adaptive spline estimators **automatically adapt** to the inhomogeneity of the underlying function.

Linear Splines and the ReLU

- If f is a linear spline, we have that

$$\mathrm{D}^2 f = \sum_{n=1}^{N} a_n \delta(\cdot - t_n).$$

- f can be written as

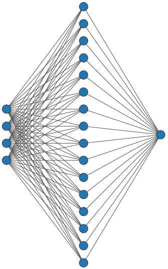
$$f(x) = \sum_{n=1}^{N} a_n \, \mathrm{ReLU}(x - t_n) + c_1 x + c_0$$

$\implies \quad \mathrm{D}^2 \, \mathrm{ReLU} = \delta.$

- The ReLU is the building block of linear splines!

Shallow Neural Networks

Shallow neural networks are are functions $f : \mathbb{R}^d \to \mathbb{R}$ that can be written as



$$f(\boldsymbol{x}) = \boldsymbol{v}^\mathsf{T} \boldsymbol{\rho}(\mathbf{W}\boldsymbol{x} - \boldsymbol{b}) = \sum_{n=1}^{N} v_n \rho(\boldsymbol{w}_n^\mathsf{T} \boldsymbol{x} - b_n),$$

where $v_n \in \mathbb{R}$, $\boldsymbol{w}_n \in \mathbb{R}^d$, $b_n \in \mathbb{R}$, and $\rho = \mathrm{ReLU}$.

Observation

When $d = 1$, we have

$$f_{\boldsymbol{v},\boldsymbol{w},\boldsymbol{b},\boldsymbol{c}}(x) = \sum_{n=1}^{N} v_n \rho(w_n x - b_n) + c_1 x + c_0$$

This is a linear spline with N knots!

- $\mathrm{D}^2 f_{\boldsymbol{v},\boldsymbol{w},\boldsymbol{b},\boldsymbol{c}} = \sum_{n=1}^{N} v_n |w_n| \delta(\cdot - b_n / w_n)$.

Observation

The solutions to the **neural network training problem**

$$\min_{\boldsymbol{\theta}=(\boldsymbol{v},\boldsymbol{w},\boldsymbol{b},\boldsymbol{c})} \sum_{m=1}^{M} \ell(y_m, f_{\boldsymbol{\theta}}(x_m)) + \lambda \underbrace{\sum_{n=1}^{N} |v_n||w_n|}_{=\mathrm{TV}^2(f_{\boldsymbol{\theta}})}$$

solve the linear locally adaptive spline variational problem

$$\min_{f \in \mathrm{BV}^2(\mathbb{R})} \sum_{m=1}^{M} \ell(y_m, f_{\boldsymbol{\theta}}(x_m)) + \lambda \, \mathrm{TV}^2(f)$$

so long as $N \geq M$.

Neural Network Training

- Neural networks are often trained with **weight decay**:

$$\min_{\boldsymbol{\theta}=(\boldsymbol{v},\boldsymbol{w},\boldsymbol{b},\boldsymbol{c})} \sum_{m=1}^{M} \ell(y_m, f_{\boldsymbol{\theta}}(x_m)) + \frac{\lambda}{2} \sum_{n=1}^{N} |v_n|^2 + |w_n|^2$$

- For any $\gamma > 0$, the mapping $(v_n, w_n) \mapsto (v_n/\gamma, \gamma w_n)$ does not change the function $f_{\boldsymbol{\theta}}$ since the ReLU is 1-homogeneous.

$\Longrightarrow$ At the solution $|v_n| = |w_n|$.

- Training a neural network with weight decay is equivalent to

$$\min_{\boldsymbol{\theta}=(\boldsymbol{v},\boldsymbol{w},\boldsymbol{b},\boldsymbol{c})} \sum_{m=1}^{M} \ell(y_m, f_{\boldsymbol{\theta}}(x_m)) + \lambda \sum_{n=1}^{N} |v_n||w_n|$$

- This observation was, first made in the 1990s. (Grandvalet 1998)

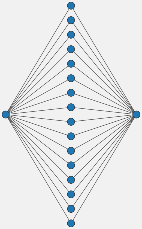
$\Longrightarrow$ Popularized recently. (Neyshabur et al. 2015)

Neural Networks and Locally Adaptive Splines

Shallow, univariate ReLU networks **trained with weight decay** are linear locally adaptive splines. (Savarese et al. 2019)

Observations (P. & Nowak 2020)

- Shallow, univariate ReLU networks learn functions in the **Banach space** BV^2.
- Shallow, univariate ReLU neural networks need to be critically parameterized or overparameterized ($N \geq M$ suffices).



Rahul Parhi and Robert D. Nowak (2020). "The Role of Neural Network Activation Functions". In: **IEEE Signal Processing Letters** 27, pp. 1779–1783. DOI: 10.1109/LSP.2020.3027517.

Shallow Multivariate Neural Networks

- In the univariate case, D^2 is a **sparsifying transform** for ReLU neurons

$$\begin{aligned} \mathrm{D}^2 \rho(wx - b) &= \mathrm{D}\, w\, u(wx - b) \\ &= w^2 \delta(wx - b) \\ &= |w| \delta(x - b/w). \end{aligned}$$

- Multivariate neurons take the form $\boldsymbol{x} \mapsto \rho(\boldsymbol{w}^\mathsf{T}\boldsymbol{x} - b)$, $\boldsymbol{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$.

  $\Longrightarrow$ These are **ridge functions**.

Question

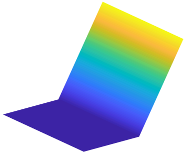
Is there an operator that sparsifies a multivariate neuron?

Answer

Yes, and it involves the Radon transform.

The Radon Transform

- Ridge functions (plane waves) are univariate functions **extended** outward in all other dimenions. Consider $\mathrm{ReLU}(x) = x_+$.



- We can use the Radon transform of a function $f : \mathbb{R}^d \to \mathbb{R}$,

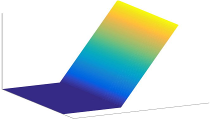
$$\mathscr{R}\{f\}(\boldsymbol{\alpha}, t) = \int_{\mathbb{R}^d} f(\boldsymbol{x}) \delta(\boldsymbol{\alpha}^\mathsf{T} \boldsymbol{x} - t) \, \mathrm{d}\boldsymbol{x}, \quad (\boldsymbol{\alpha}, t) \in \mathbb{S}^{d-1} \times \mathbb{R},$$

  to "extract" the underlying univariate function to extend results for univariate functions to multivariate ridge functions.

The Sparsifying Operator

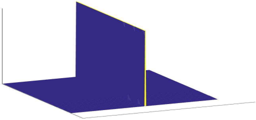
- ReLU Neuron

$$\Longrightarrow \quad \rho(\boldsymbol{w}_0^{\mathsf{T}}(\cdot) - b_0), \; (\boldsymbol{w}_0, b_0) \in \mathbb{S}^{d-1} \times \mathbb{R}$$



- **Laplacian** of neuron

$$\Longrightarrow \quad \Delta\big\{\rho(\boldsymbol{w}_0^{\mathsf{T}}(\cdot) - b_0)\big\} = \delta(\boldsymbol{w}_0^{\mathsf{T}}(\cdot) - b_0)$$



- **Filtered Radon transform** of Laplacian of neuron[1]

$$\Longrightarrow \quad (\mathrm{K}^{d-1} \mathscr{R} \Delta)\big\{\rho(\boldsymbol{w}_0^{\mathsf{T}}(\cdot) - b_0)\big\}(\boldsymbol{\alpha}, t) = \delta_{\mathscr{R}}((\boldsymbol{\alpha}, t) - (\boldsymbol{w}_0, b_0)).$$

- This operator has gained popularity due to the seminal work of Ongie et al. (2020).

[1] Kurková et al. 1997; Ongie et al. 2020; P. & Nowak 2021; Unser 2022

“Native Space” for Shallow Neural Networks

Question

What would be a multivariate analogue of $\mathrm{BV}^2(\mathbb{R})$?

Answer

$\mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d)$, the second-order Radon-domain BV space.

$$\mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d) \coloneqq \left\{ f : \mathbb{R}^d \to \mathbb{R} : \; \mathscr{R}\,\mathrm{TV}^2(f) < \infty \right\}$$

- $\mathscr{R}\,\mathrm{TV}^2(f) \coloneqq \|\mathrm{K}^{d-1}\,\mathscr{R}\,\Delta f\|_{\mathcal{M}}$
  $\implies \; \mathrm{TV}^2(f) = \|\mathrm{D}^2 f\|_{\mathcal{M}}$.
- When $d = 1$,
  $\implies \; \mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d) = \mathrm{BV}^2(\mathbb{R})$ and $\mathscr{R}\,\mathrm{TV}^2(\cdot) = \mathrm{TV}^2(\cdot)$.
- $\mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d)$ is a **Banach space**. (P. & Nowak 2021)

A Representer Theorem for Shallow Neural Networks

Theorem (P. & Nowak 2021)

Consider the variational problem

$$f_{\mathrm{ReLU}} \in \operatorname*{arg\,min}_{f \in \mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d)} \sum_{m=1}^{M} \ell(y_m, f(\boldsymbol{x}_m)) + \lambda \mathscr{R}\,\mathrm{TV}^2(f),$$

- Solution set is nonempty, convex, and weak* compact.
- **Extreme points** of solution set take the form

$$f_{\mathrm{ReLU}}(\boldsymbol{x}) = \sum_{n=1}^{N} v_n \rho(\boldsymbol{w}_n^{\mathsf{T}} \boldsymbol{x} - b_n) + \boldsymbol{c}^{\mathsf{T}} \boldsymbol{x} + c_0, \quad N < M$$

- Full solution set is convex hull of extreme points.

$\Longrightarrow$ Solution set is completely characterized by ReLU networks.

Rahul Parhi and Robert D. Nowak (2021). "Banach space representer theorems for neural networks and ridge splines". In: **Journal of Machine Learning Research** 22.43, pp. 1–40.

Neural Network Training

The solutions to the **neural network training problem**

$$\min_{\boldsymbol{\theta}} \sum_{m=1}^{M} \ell(y_m, f_{\boldsymbol{\theta}}(\boldsymbol{x}_m)) + \frac{\lambda}{2} \sum_{n=1}^{N} |v_n|^2 + \|\boldsymbol{w}_n\|_2^2$$

solve the variational problem

$$\min_{f \in \mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d)} \sum_{m=1}^{M} \ell(y_m, f(\boldsymbol{x}_m)) + \lambda\, \mathscr{R}\,\mathrm{TV}^2(f).$$

so long as $N \geq M$.

Observation

Shallow, multivariate ReLU networks learn functions in the **Banach space** $\mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d)$.

Rahul Parhi and Robert D. Nowak (2021). "Banach space representer theorems for neural networks and ridge splines". In: **Journal of Machine Learning Research** 22.43, pp. 1–40.

Deep Neural Networks

Consider the cascade of $\mathscr{R}\,\mathrm{BV}^2$ spaces:

$$\mathscr{R}\,\mathrm{BV}^2_{\mathsf{deep}} = \left\{ f = f^{(L)} \circ \cdots \circ f^{(1)} \,:\, f^{(\ell)} \in \mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^{d_{\ell-1}}; \mathbb{R}^{d_\ell}) \right\}$$

Theorem (P. & Nowak 2022)

There exists a solution to the variational problem

$$\min_{f \in \mathscr{R}\,\mathrm{BV}^2_{\mathsf{deep}}} \sum_{m=1}^{M} \ell(y_m, f(\boldsymbol{x}_m)) + \lambda \sum_{\ell=1}^{L} \mathscr{R}\,\mathrm{TV}^2(f^{(\ell)}),$$

- that takes the form a deep ReLU neural network.
  - $\Longrightarrow$ with L hidden layers
  - $\Longrightarrow$ linear bottlenecks
  - $\Longrightarrow$ **sparse** solutions (widths $O(M^2)$)

Rahul Parhi and Robert D. Nowak (2022b). "What kinds of functions do deep neural networks learn? Insights from variational spline theory". In: **SIAM Journal on Mathematics of Data Science** 4.2, pp. 464–489.

What is $\mathscr{R}\,\mathrm{BV}^2(\Omega)$?

- Let $\Omega = \left\{\boldsymbol{x} \in \mathbb{R}^d : \ \|\boldsymbol{x}\| \leq 1\right\}$. Then,

$$\mathscr{R}\,\mathrm{BV}^2(\Omega) \coloneqq \{f : \Omega \to \mathbb{R} : \ \exists g \in \mathscr{R}\,\mathrm{BV}^2(\mathbb{R}^d) \text{ s.t. } g\big|_{\Omega} = f\}$$

- Every $f \in \mathscr{R}\,\mathrm{BV}^2(\Omega)$ admits an **integral representation**

$$f(\boldsymbol{x}) = \int_{\mathbb{S}^{d-1}\times[-1,1]} \rho(\boldsymbol{w}^\mathsf{T}\boldsymbol{x} - b)\,\mathrm{d}\mu(\boldsymbol{w}, b) + \boldsymbol{c}^\mathsf{T}\boldsymbol{x} + c_0$$

$\Longrightarrow$ P. & Nowak 2022

- Such integral representations have been studied for a number of years and are referred to as the **variation spaces** of shallow neural networks. (Kurková and Sanguineti 2001; Mhaskar 2004; Bach 2017; Siegel and Xu 2021a)
- Our work provides an **analytic characterization** of these variation spaces.

Neural Spaces

- The **spectral Barron spaces** $\mathscr{B}^s(\mathbb{R}^d)$ are defined via the norm

$$\|f\|_{\mathscr{B}^s(\mathbb{R}^d)} = \|(1 + \|\cdot\|_2)^s \hat{f}(\cdot)\|_{\mathcal{M}} = \int_{\mathbb{R}^d} (1 + \|\boldsymbol{\omega}\|_2)^s |\hat{f}(\boldsymbol{\omega})| \, \mathrm{d}\boldsymbol{\omega}$$

$\Longrightarrow$ Proposed in the seminal work of Barron on the approximation properties of shallow neural networks. (Barron 1993)

- On a bounded domain Ω, we have for any $\varepsilon > 0$,

$$H^{d/2+2+\varepsilon}(\Omega) \overset{\text{c.}}{\hookrightarrow} \mathscr{B}^2(\Omega) \overset{\text{c.}}{\hookrightarrow} \mathscr{R}\,\mathrm{BV}^2(\Omega) \overset{\text{c.}}{\hookrightarrow} L^2(\Omega)$$

$\Longrightarrow$ $H^s(\Omega)$ is the sth-order L^2-Sobolev space on Ω.

$\Longrightarrow$ Klusowski and Barron 2018; Xu 2020; Siegel and Xu 2021; P. and Nowak 2022

Approximation Properties of $\mathscr{R}\,\mathrm{BV}^2(\Omega)$

- It is well-known how to approximate integrals of the form

$$\int_{\mathbb{S}^{d-1}\times[-1,1]} \rho(\boldsymbol{w}^\mathsf{T}\boldsymbol{x} - b)\,\mathrm{d}\mu(\boldsymbol{w}, b)$$

$\Longrightarrow$ Maurey and Pisier 1981; Barron 1993; Matoušek 1996; Siegel and Xu 2021

- Given $f \in \mathscr{R}\,\mathrm{BV}^2(\Omega)$, there exists a shallow neural network f_N with N neurons such that

$$\|f - f_N\|_{L^2} \lesssim N^{-\frac{1}{2}-\frac{3}{2d}} \lesssim N^{-\frac{1}{2}}$$

$\Longrightarrow$ This rate does not grow with the input dimension d.
$\Longrightarrow$ Shallow neural networks **break the curse of dimensionality**.

- Compare with approximation in $H^2[0,1]^d$. The best N term L^2-approximation rate is $N^{-\frac{2}{d}}$ (use a truncated Fourier series), which suffers the curse of dimensionality.

Estimation Properties of $\mathscr{R}\,\mathrm{BV}^2(\Omega)$

- Given $f \in \mathscr{R}\,\mathrm{BV}^2(\Omega)$, suppose we observe

$$y_m = f(\boldsymbol{x}_m) + \varepsilon_m,\ m = 1, \dots, M,$$

where $\{\boldsymbol{x}_m\}_{m=1}^M \subset \Omega$ are nicely distributed and $\{\varepsilon_m\}_{m=1}^M$ are i.i.d. white noise.

- Any solution to the neural network training problem

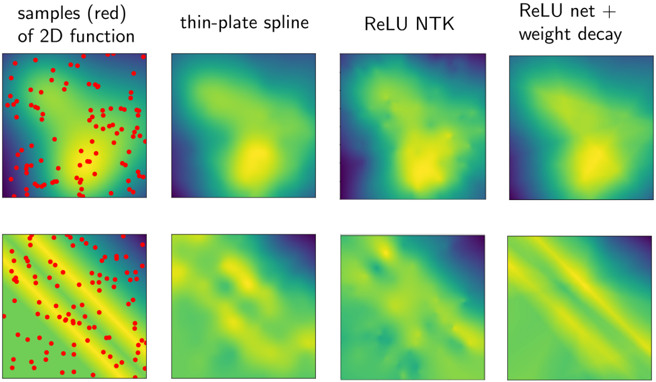
$$f_M \in \arg\min_{\boldsymbol{\theta}} \sum_{m=1}^{M} |y_m - f_{\boldsymbol{\theta}}(\boldsymbol{x}_m)|^2 + \frac{\lambda}{2}\sum_{n=1}^{N} |v_n|^2 + \|\boldsymbol{w}_n\|_2^2$$

satisfies

$$\mathbb{E}\|f - f_M\|_{L^2}^2 \lesssim M^{-\frac{d+3}{2d+3}} \lesssim M^{-\frac{1}{2}}.$$

$\Longrightarrow$ This rate does not grow with the input dimension d.
$\Longrightarrow$ This is the **minimax rate**. (P. & Nowak, 2021)

Data Fitting and Extrapolation



neural networks learn and extrapolate very differently than classical multivariate estimation techniques and kernel methods in general

Linear methods necessarily suffer the curse of dimensinality when estimating $\mathscr{R}\,\mathrm{BV}^2(\Omega)$ functions from data.

- Minimax lower bound for linear methods: $M^{-\frac{3}{d+3}}$. (P. & Nowak, 2022)

$\mathscr{R}\,\mathrm{BV}^2(\Omega)$ is a Mixed Variation Space

- Functions in $\mathscr{R}\,\mathrm{BV}^2$ can be very smooth in all directions (e.g., in the Sobolev space $H^{d/2+2+\varepsilon}$.
- Functions in $\mathscr{R}\,\mathrm{BV}^2$ can be very nonsmooth in all but a few directions (e.g., a ridge function with a piecewise linear profile).
- Such spaces are referred to as "mixed variation" spaces. (Donoho 2000)

The Fundamental Questions

- What kinds of functions do neural networks learn?
  - $\Longrightarrow$ ReLU networks trained with weight decay are optimal solutions to variational problems over $\mathscr{R}\,\mathrm{BV}^2$-type **Banach spaces**.
- Why can neural networks perform well in high dimensional settings?
  - $\Longrightarrow$ Dimension-free approximation and estimation rates.
- What is the right way to regularize a neural network?
  - $\Longrightarrow$ Radon-domain total variation $\iff$ weight decay.

Concluding Remarks

- The $\mathscr{R}\,\mathrm{BV}^2$ function space perspective of neural network provides a **concrete framework** to compare neural networks to classical data-fitting techniques such as kernel methods.
- Many researchers study infinite-width neural networks.

  $\Longrightarrow$ Our **representer theorems** say there is no need to consider networks of arbitrary width.
- **Skip connections** are often used in network architectures.

  $\Longrightarrow$ They are a natural by-product of the variational formulation of the learning problem.
- One paradigm for understanding neural networks is through the **neural tangent kernel** (i.e., assuming the problem is over a Hilbert space).

  $\Longrightarrow$ $\mathscr{R}\,\mathrm{BV}^2$ is a non-Hilbertian **Banach space**.

References I

Bach, Francis (2017). "Breaking the curse of dimensionality with convex neural networks". In: **The Journal of Machine Learning Research** 18.1, pp. 629–681.

Barron, Andrew R. (1993). "Universal approximation bounds for superpositions of a sigmoidal function". In: **IEEE Transactions on Information theory** 39.3, pp. 930–945.

de Boor, Carl and Robert E. Lynch (1966). "On splines and their minimum properties". In: **Journal of Mathematics and Mechanics** 15.6, pp. 953–969.

Donoho, David L. (2000). "High-dimensional data analysis: The curses and blessings of dimensionality". In: **AMS math challenges lecture** 1.2000, p. 32.

Donoho, David L. and Iain M. Johnstone (1998). "Minimax estimation via wavelet shrinkage". In: **The Annals of Statistics** 26.3, pp. 879–921.

Fisher, Stephen D. and Joseph W. Jerome (1975). "Spline solutions to L^1 extremal problems in one and several variables". In: **Journal of Approximation Theory** 13.1, pp. 73–83.

Grandvalet, Yves (1998). "Least absolute shrinkage is equivalent to quadratic penalization". In: **International Conference on Artificial Neural Networks**. Springer, pp. 201–206.

Kimeldorf, George S. and Grace Wahba (1970). "A correspondence between Bayesian estimation on stochastic processes and smoothing by splines". In: **The Annals of Mathematical Statistics** 41.2, pp. 495–502.

Klusowski, Jason M. and Andrew R. Barron (2018). "Approximation by Combinations of ReLU and Squared ReLU Ridge Functions With ℓ^1 and ℓ^0 Controls". In: **IEEE Transactions on Information Theory** 64.12, pp. 7649–7656.

References II

Kurková, Věra, Paul C Kainen, and Vladik Kreinovich (1997). "Estimates of the number of hidden units and variation with respect to half-spaces". In: **Neural Networks** 10.6, pp. 1061–1068.

Kurková, Vera and Marcello Sanguineti (2001). "Bounds on rates of variable-basis and neural-network approximation". In: **IEEE Transactions on Information Theory** 47.6, pp. 2659–2665.

Mammen, Enno and Sara van de Geer (1997). "Locally adaptive regression splines". In: **The Annals of Statistics** 25.1, pp. 387–413.

Matoušek, Jiří (1996). "Improved upper bounds for approximation by zonotopes". In: **Acta Mathematica** 177.1, pp. 55–73.

Mhaskar, Hrushikesh N. (2004). "On the tractability of multivariate integration and approximation by neural networks". In: **Journal of Complexity** 20.4, pp. 561–590.

Neyshabur, Behnam, Ryota Tomioka, and Nathan Srebro (2015). "In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning.". In: **International Conference on Learning Representations (Workshop)**.

Ongie, Greg, Rebecca Willett, Daniel Soudry, and Nathan Srebro (2020). "A Function Space View of Bounded Norm Infinite Width ReLU Nets: The Multivariate Case". In: **International Conference on Learning Representations**.

Parhi, Rahul and Robert D. Nowak (2020). "The Role of Neural Network Activation Functions". In: **IEEE Signal Processing Letters** 27, pp. 1779–1783. DOI: 10.1109/LSP.2020.3027517.

— (2021). "Banach space representer theorems for neural networks and ridge splines". In: **Journal of Machine Learning Research** 22.43, pp. 1–40.

References III

Parhi, Rahul and Robert D. Nowak (2022a). "Near-minimax optimal estimation with shallow ReLU neural networks". In: **IEEE Transactions on Information Theory**.

— (2022b). "What kinds of functions do deep neural networks learn? Insights from variational spline theory". In: **SIAM Journal on Mathematics of Data Science** 4.2, pp. 464–489.

Savarese, Pedro, Itay Evron, Daniel Soudry, and Nathan Srebro (2019). "How do infinite width bounded norm networks look in function space?" In: **Conference on Learning Theory**. PMLR, pp. 2667–2690.

Siegel, Jonathan W. and Jinchao Xu (2021a). "Characterization of the Variation Spaces Corresponding to Shallow Neural Networks". In: **arXiv preprint arXiv:2106.15002**.

— (2021b). "Sharp Bounds on the Approximation Rates, Metric Entropy, and n-widths of Shallow Neural Networks". In: **arXiv preprint arXiv:2101.12365v7**.

Unser, Michael, Julien Fageot, and John Paul Ward (2017). "Splines Are Universal Solutions of Linear Inverse Problems with Generalized TV Regularization". In: **SIAM Review** 59.4, pp. 769–793.

Xu, Jinchao (2020). "Finite neuron method and convergence analysis". In: **Communications in Computational Physics** 28.5, pp. 1707–1745.